Simulation of Lean Premixed Turbulent Combustion

J. Bell[1], M. Day[1], A. Almgren[1], M. Lijewski[1], C. Rendleman[1], R. Cheng[2], I. Shepherd[2]

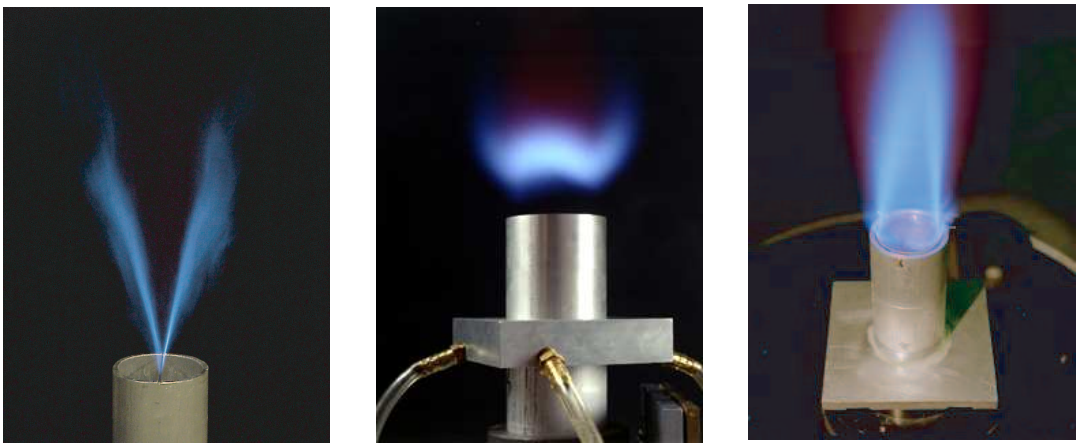
[1] CCSE, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA
[2] EETD, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

E-mail: jbbell@lbl.gov

Abstract. There is considerable technological interest in developing new fuel-flexible combustion systems that can burn fuels such as hydrogen or syngas. Lean premixed systems have the potential to burn these types of fuels with high efficiency and low NOx emissions due to reduced burnt gas temperatures. Although traditional scientific approaches based on theory and laboratory experiment have played essential roles in developing our current understanding of premixed combustion, they are unable to meet the challenges of designing fuel-flexible lean premixed combustion devices. Computation, with its ability to deal with complexity and its unlimited access to data, has the potential for addressing these challenges. Realizing this potential requires the ability to perform high fidelity simulations of turbulent lean premixed flames under realistic conditions. In this paper, we examine the specialized mathematical structure of these combustion problems and discuss simulation approaches that exploit this structure. Using these ideas we can dramatically reduce computational cost, making it possible to perform high-fidelity simulations of realistic flames. We illustrate this methodology by considering ultra-lean hydrogen flames and discuss how this type of simulation is changing the way researchers study combustion.

1. Introduction

Concerns over U.S. dependence on imports of foreign oil, coupled with pollution and greenhouse gas emission issues, have generated significant interest in alternative fuels such as hydrogen or syngas, which is obtained from coal gasification. Effective utilization of these fuels requires combustion devices that can operate cleanly and efficiently over a broad range of fuels and fuel mixtures. Lean-premixed systems have the potential for meeting these requirements; they operate at high efficiency and have low NOx emissions due to reduced burnt gas temperatures. However, it is difficult to design safe and reliable systems based on premixed combustion. Unlike non-premixed systems where the flame location and stability are controlled by the mixing of fuel and oxidizer, premixed flames require device-scale stabilization techniques to create a stable and statistically stationary flame. The large-scale structure and behavior of the flame is determined by its coupling with the flow field across a broad range of scales ranging from the device scale and the dominant energy-bearing scales down to scales associated with the turbulent energy cascade. Several stabilization mechanisms are considered in the literature [1]. For example, the Twenty-Ninth Combustion Symposium includes studies by Sattler et al. [2] of a turbulent V-flame, Shepherd et al. [3] of a swirl-stabilized flame, Most et al. [4] of a bluff-body stabilized flame, and Chen et al. [5] of Bunsen and stagnation flames. Examples of several stabilized premixed flames are shown in Figure 1.



V-Flame Low-swirl burner Envelope Flame

Figure 1. Three different premixed methane flame with different stabilization mechanisms.

Traditional approaches to the study of turbulent premixed combustion are based on a theoretical model for premixed flames that treats the flame as an interface propagating through the fluid at a speed determined by the internal flame structure given by one-dimensional flame analysis. (See, for example, Peters [6].) In its simplest form, this interface propagates at the one-dimensional laminar flame speed; the overal rate at which the turbulent flame consumes fuel is determined by the degree to which turbulence wrinkles the flame front, increasing its surface area and so enhancing the burning rate. More elaborate models incorporate modifications to the local flame speed based based on the flame stretch due to the curvature of the front and the local fluid strain field. In essence this "thin flame" approach assumes a separation between the chemical scales (the flame front) and the significant turbulence scales in the flow. This basic model has proven highly successful for natural gas combustion where it forms the basis for phenomenological turbulence/chemistry interaction models in engineering analysis codes and provides the context for the analysis of experimental data. (See Bray [7] for a discussion of how experimental data are used in conjunction with closure models for engineering design studies.)

Unfortunately, this traditional approach breaks down at a fundamental level when applied to the lean premixed combustion of some alternative fuels. The origin of the difficulties is illustrated in Figure 2, which show the flame surface in an ultra-lean H_2 flame at $\phi = 0.27$. In Figure 2a we show a Mie scattering image that uses small oil droplets which evaporate at

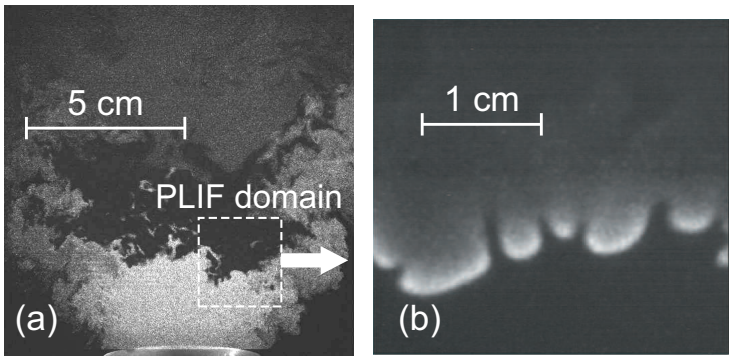


Figure 2. Experimental data taken from a lean ($\phi = 0.27$) hydrogen flame in a low swirl burner. (a) Mie-scattering image indicating instantaneous flame surface morphology, and (b) OH PLIF over a smaller region of a similar flame, indicating the cellular structure of the burning regions, with gaps in the flame representing local extinction.

650K to visualize the position of the instantaneous flame front. In Figure 2b we show a laser induced fluoresence (PLIF) image that shows where the OH radical, a key intermediate in H_2 combustion, is concentrated. In this image, we see clear breaks in the flame suggesting local extinction of the flame. (This type of behavior, referred to as cellular structure, has been observed both experimentally and computationally in hydrogen combustion, see [8, 9].) We no longer have a continuous flame surface; the flame cannot accurately represented as a continous interface propagating through the domain. Thus, the traditional approach to studying premixed flames is not applicable here. Consequently, standard approaches to analysis of experimental data and conventional engineering design methodologies cannot be used.

Computation has the potential to play a key role in understanding the dynamics of these types of lean premixed flames and providing the insight needed to analyze experimental data and develop new engineering design methodologies. Simulation needs to meet several requirements to fulfill this role. First it must be able to simulate a sufficiently large range of scales to represent the types of turbulent conditions found in laboratory experiments while adequately resolving the internal structure of the flame. Typical laboratory experiments are characterized by an integral scale $\ell_t \approx 2 - 6$ mm. We would like to simulate domains that are on the order 10 ℓ_t to provide a realistic range of turbulence scales, i.e., domains that are several centimeters on each side. The second requirement is that the simulations incorporate realistic models for chemistry and transport to provide the necessary physics fidelity to represent the interplay of turbulence and chemistry.

In this paper, our goal is to discuss the steps in developing a simulation capability that meets these requirements. We identify three major components in the development of computational methodology:

- **Mathematical model**
  - How do we translate the physical problem we wish to solve into a mathematical problem?
  - Is the model well-posed?
  - What are the properties of solutions of the model?
- **Discretization**
  - How do we approximate the mathematical model with a finite number of degrees of freedom?
  - What are the accuracy and stability properties of the discretization?
- **Solution algorithms and software**
  - How do we solve the resulting discrete systems of equations computationally?
  - How do we implement the overall algorithm?
  - Does the implementation efficiently solve the system?
  - Does the software effectively utilize computing resources?

In the remainder of the paper we illustrate these steps in defining a computational methodology for premixed combustion and illustrate the methodology on ultra-lean H_2 combustion. We note that there is considerable interplay between these various components; the overall philosophy here is based on considering the process in its entirety.

2. Mathematical model

The starting point for the development of the mathematical model is the compressible Navier-Stokes equations for a reacting flow. Specifically, we consider a multicomponent gaseous mixture ignoring Soret and Dufour effects, body forces and radiative heat transfer, and assume a mixture model for species diffusion[10, 11]. The governing equations, which express conservation of mass, momentum and energy, augmented by species transport, are

$$
\begin{aligned}
\frac{\partial \rho}{\partial t} + \nabla \cdot \rho U &= 0, \\
\frac{\partial \rho U}{\partial t} + \nabla \cdot \rho U U + \nabla p &= \nabla \cdot \tau, \\
\frac{\partial \rho E}{\partial t} + \nabla \cdot \rho U E + p U &= \nabla \cdot \lambda \nabla T + \sum_m \nabla \cdot h_m \rho D_m \nabla Y_m + \nabla \cdot \tau U, \\
\frac{\partial \rho Y_m}{\partial t} + \nabla \cdot \rho U Y_m &= \nabla \cdot \rho D_m \nabla Y_m - \dot{\omega}_m,
\end{aligned}
$$

where ρ is the density, U is the velocity, $E = \Sigma e_m(T) Y_m + 1/2 U \cdot U$ is the total energy, Y_m is the mass fraction of species m, T is the temperature, and $\dot{\omega}_m$ is the net destruction rate for species m due to chemical reactions. Also, λ is the thermal conductivity, τ is the stress tensor, c_p is the specific heat of the mixture, and $e_m(T), h_m(T)$, and D_m are the internal energy, enthalpy and species mixture-averaged diffusion coefficients of species m, respectively. We note that for these equations, we have $\Sigma Y_m = 1$, $\Sigma \rho D_m \nabla Y_m = 0$, and $\Sigma \dot{\omega}_m = 0$ so that the sum of the species transport equations gives the conservation of total mass. These evolution equations are supplemented by an equation of state for a perfect gas mixture:

$$
p_0 = \rho R_{mix} T = \rho \mathcal{R} T \sum_m \frac{Y_m}{W_m}
$$

where W_m is the molecular weight of species m, and $\mathcal{R}$ is the universal gas constant.

One possible approach to simulating reacting flow is to simply discretize this system of equations. However, the compressible flow equations provide a general description of essentially continuum fluid dynamic phenomena. Consequently, before discretizing we first consider the basic characteristics of premixed flames and discuss how those characteristics relates to more general fluid dynamics. The compressible flow equations include two types of waves. They include the transport of material at the fluid velocity U and acoutic waves propagating at speed $U + c$ where c is the speed of sound. For typical premixed flame experiments, $U \approx 3 - 30$ m/s while the sound speed in the hot product gases is about 1000 m/s. Thus, the time scale of the fluid motion is considerably longer than the time scale of acoustic wave propagation.

How do these time scales relate to the time scales of the flame? Typical thermal flame thickness, δ_T, is 1 mm or more for lean flames while the laminar flame speed $s_L \approx 5 - 50$ cm/s. We can define time scales for flame propagation, fluid motion and acoustic wave as δ_T / S_L, $\delta_T / |U|$ and δ_T / c, respectively. With these definitions, we see that the acoustic time scale is much smaller than the flame and fluid time scales. This type of flow is referred to as low Mach number since the Mach number, $M = |U|/c << 1$. An alternative to discretization of the compressible equations directly is to exploit the separation of scales between the acoustic waves and the fluid motion by adopting a low Mach number formulation.

Before discussing the low Mach number model, it is worth noting how the fluid dynamical time scales relate to the chemical time scales. Detailed kinetics mechanisms are typically quite stiff with time scales much faster than the fluid time scale δ_T / U. However, chemistry couples to the fluid motion primarily through heat release. Heat release time scales for the flames considered here, based on FWHM of the heat release for lean laminar flames, are 0.1-0.3 times the flame time scale. Thus, the time scale of the heat release is roughly comparable to the fluid time scale. The fast chemical time scales, which represent fast, near-equilibrium reactions, do not couple directly to the fluid. Consequently, although we must address the stiffness of the chemical reactions in the numerics, they do not pose a problem for adopting a low Mach number formulation. (We note that this characteristic is a property of the type of problems we are

considering; in other situations, such as detonation phyics, energy is released on acoustic time scales and a compressible formulation is needed.)

The low Mach number combustion formulation was first introduced by Rehm and Baum [12] and was later derived rigorously from low Mach number asymptotic analysis by Majda and Sethian [13]. The basic steps of the analysis are first to normalize the problem by rescaling and then to expand the terms in the compressible Navier-Stokes equations in M. Equating terms in Mach number and examing the behavior as $M \to 0$ one can show that in an unconfined domain, the pressure can be decomposed as

$$p(x,t) = p_0 + \pi(x,t)$$

where p_0 is the ambient thermodynamic pressure and π is a perturbational pressure field that satisfies $\pi/p_0 \sim \mathcal{O}\left(M^2\right)$. (In a more general setting, p_0 is a function of t.) With this decomposition, p_0 defines the thermodynamic state; all thermodynamic quantities are independent of π.

With this decomposition, the low Mach number equations for an open domain are

$$\begin{aligned}
\frac{\partial \rho}{\partial t} + \nabla \cdot \rho U &= 0, \\
\frac{\partial \rho U}{\partial t} + \nabla \cdot \rho U U + \nabla \pi &= \nabla \cdot \tau, \\
\frac{\partial \rho h}{\partial t} + \nabla \cdot \rho U h &= \nabla \cdot \lambda \nabla T + \sum_m \nabla \cdot h_m \rho D_m \nabla Y_m, \\
\frac{\partial \rho Y_m}{\partial t} + \nabla \cdot \rho U Y_m &= \nabla \cdot \rho D_m \nabla Y_m - \dot{\omega}_m,
\end{aligned}$$

Although similar to the compressible equations, the low Mach number model expresses the energy equation in terms of enthalpy, $h(T, Y_m) = \sum_m h_m(T) Y_m$. More importantly, acoustic waves are instanteously equilibrated and the equation of state, $p_0 = \rho R_{mix} T$, now constrains the evolution. As a result, this description retains compressiblity effect due to heat release but removes the time scale associated with acoustic wave propagation from the dynamics of the system. In this form, the perturbation pressure, π, plays the role of a Lagrange multiplier to constrain the evolution so that this constraint is satisfied.

Although not directly needed to specify the model, by differentiating the enthalpy equation we can derive an auxiliary equation for temperature of the form

$$\rho c_p \Big(\frac{\partial T}{\partial t} + U \cdot \nabla T \Big) = \nabla \cdot \lambda \nabla T + \sum_m \Big(\rho D_m \nabla Y_m \cdot \nabla h_m + h_m \dot{\omega}_m \Big)$$

3. Discretization methodology

By adopting a low Mach number formulation, we have analytically removed acoustic wave propagation from the system. This allows us to construct a temporal discretization approach that advances the solution at the fluid time scales rather than the acoustic time scales. However, in changing the form of the equations, we have also changed to system from an initial value problem to a system that evolves subject to a constraint, referred to as a differential algebraic equation (DAE). DAE's are typically harder to treat numerically than initial value problems. See [14, 15] for a discussion of discretization procedures for DAE's.

DAE's are classified based on their index, which is the minimum number of times the constraint needs to be differentiated to reduce the problem to an intial value problem. In essence, the higher the index, the harder the system is to solve. (The low Mach number equations are

index 3.) One approach to solving a constrained system is to differentiate the constraint, referred to as index reduction. In the context of low Mach number combustion, we differentiate the EOS along particle paths and use the evolution equations for ρ, Y_m and the auxiliary equation for T to define a constraint on the velocity:

$$\begin{aligned}
\nabla \cdot U &= \frac{1}{\rho}\frac{D\rho}{Dt} = -\frac{1}{T}\frac{DT}{Dt} - \frac{\mathcal{R}}{R}\sum_m \frac{1}{W_m}\frac{DY_m}{Dt} \\
&= \frac{1}{\rho c_p T}\left(\nabla\cdot(\lambda\nabla T) + \sum_m \rho D_m \nabla Y_m \cdot \nabla h_m\right) + \frac{1}{\rho}\sum_m \frac{W}{W_m}\nabla(\rho D_m \nabla Y_m) \\
&\qquad + \frac{1}{\rho}\sum_m \left(\frac{W}{W_m} - \frac{h_m(T)}{c_p T}\right)\dot{\omega_m} \\
&\equiv S
\end{aligned}$$

Structurally, the low Mach number equations thus take the form

$$\rho_t + \nabla\cdot\rho u = 0$$

$$U_t + U\cdot\nabla U + \frac{1}{\rho}\nabla\pi = \frac{1}{\rho}\nabla\cdot\tau$$

$$\nabla\cdot U = S$$

with additional species and enthalpy equations whose spatial derivatives define the velocity constraint S.

The structure of the equations after differentiating the constraint is similar to the constant-density incompressible Navier Stokes equation For incompressible flows, projection-based fractional step methods, which parallel standard DAE methodologies, have proven to provide an efficient discretization strategy [16–18] Our goal then is to define a generalized projection methodology for low Mach number reacting flows. This generalization requires that we address two key differences between the incompressible flow equations and the low Mach number system. First, the low Mach number system includes finite amplitude density variations; and, second, the constraint on the velocity field is inhomogeneous. Two different projection-based sequential algorithms have been proposed. One of these approaches, developed by McMurtry *et al* [19] and Rutland and Ferziger [20], advances the thermodynamic variables and then uses the conservation of mass equation to constrain the evolution. Imposing the constraint in this form requires the solution of a Poisson equations to impose the constraint. Although this approach does not fit within a mathematical projection framework, is has been successfully used by a number of authors to model reacting flows. See, for example, [21–25].

Here, we consider a different approach based on a generalized projection framework first introduced in Bell and Marcus [26]. The basic idea here is that, subject to boundary conditions, any vector field, V can be decomposed as

$$V = U_d + \frac{1}{\rho}\nabla\phi$$

where U_d is divergence free and U_d and $\frac{1}{\rho}\nabla\phi$ are orthogonal with a suitable inner product. In particular, this decomposition is an exact analog to the standard Hodge decomposition in a ρ-weighted inner product; e.g.,

$$\int (U_d \cdot \frac{1}{\rho}\nabla\phi)\rho\, dm = 0$$

Using this inner product, we can define a ρ-based projection, $\mathbf{P}_\rho$ such that $\mathbf{P}_\rho V = U_d$ with $||\mathbf{P}_\rho|| = 1$ and $\mathbf{P}_\rho^2 = \mathbf{P}_\rho$.

This projection operator allows us to address the finite amplitude density variations in the low Mach number system. The Majda and Sethian analysis [13] shows that the compressibility effects in the flow, given by S, can be represented in terms of the gradient of a potential flow,

$$\nabla \cdot \nabla \xi = S$$

Using this form, we can further generalize the vector field decomposition to write any velocity field as

$$V = U_d + \nabla \xi + \frac{1}{\rho} \nabla \phi$$

We can then define

$$U = \mathbf{P}_\rho (V - \nabla \xi) + \nabla \xi$$

so that $\nabla \cdot U = S$ and $\mathbf{P}_\rho(\frac{1}{\rho}\nabla\phi) = 0$

This construction provides the basis for a projection algorithm for the low Mach number equations. (We also note that using this generalized construction we can recast the system as a pure initial value problem, which can be used to study mathematical properties of the system.)

The basic idea of the projection algorithm is to advance the thermodynamic variables and a provisional velocity with a lagged approximation to the constraint. We then use the vector field deomposition to correct the velocity field so that it satisfies the constraint. Here, we only sketch the algorithm, for details see [27, 28].

First we advance the density

$$\frac{\rho^{n+1} - \rho^n}{\Delta t} = -\nabla \cdot (\rho U^{ADV})^{n+\frac{1}{2}} \quad ,$$

species concentrations and enthalpy,

$$\frac{\rho^{n+1}\chi^{n+1} - \rho^n \chi^n}{\Delta t} + \nabla \cdot (\rho U^{ADV} \chi)^{n+\frac{1}{2}} = D_\chi + R_\chi \quad for\ \chi = h, Y_m \quad ,$$

and compute a provisional velocity

$$\frac{U^* - U^n}{\Delta t} = -[U^{ADV} \cdot \nabla U]^{n+\frac{1}{2}} - \frac{1}{\rho^{n+\frac{1}{2}}} \nabla \pi^{n-\frac{1}{2}} + \frac{1}{\rho^{n+\frac{1}{2}}} \nabla \cdot \frac{\tau^n + \tau^*}{2} \quad .$$

For this stage we use a Crank-Nicolson discretization of the viscous terms and a specialized second-order Godunov algorithm to compute the advective derivatives. As part of the Godunov algoirthm we compute an advective velocity field U^{ADV} that is projected so that the vector field used for advection satisfies the constraint. A further operator-splitting procedure is used to advance the species equations. In particular, we first advance the chemistry by $\Delta t/2$, then advance the advection and diffusion components and finally complete the chemistry advance. This allows use to decouple the chemistry and use stiff ODE integration methodology to advance the kinetics equations.

The updated thermodynamic variables are then used to compute S^{n+1} To extract the component satisfying the divergence constraint we solve

$$\nabla \cdot \left(\frac{1}{\rho} \nabla \phi \right) = \nabla \cdot \vec{U}^* - S^{n+1}$$

for ϕ, and set

$$\pi^{n+1/2} = \pi^{n-1/2} + \phi$$

and

$$\vec{U}^{n+1} = \vec{U}^{*} - \frac{1}{\rho}\nabla\phi$$

This procedure implements the generalized vector field decomposition discussed above but exploits linearity to perform only a single elliptic solve to enforce the constraint.

We note that the algorithm presented above discretely conserves density, species (up to reactions) and enthalpy. However, because we use a differentiated form of the constraint, the method can drift off the constraint surface. To correct for this drift, we include a minor correction in the projection used to compute U^{ADV} to force the solution back toward the constraint.

Given the basic discretization scheme described above, we now examine the spatial resolution requirements for flame simulations. At the integral scales and turbulent intensities found in typical laboratory scale experiments, the inflow turbulence typically has a Kolmogorov scale, $\kappa \approx 200\mu$m. In the post-flame region, the turbulence is reduced as a result of fluid expansion through the flame and increased viscosity at higher temperatures.

Typical resolution requirements suggest a spatial resolution of a small mulitple of κ. Near the flame, we need to resolve the internal structure of the flame, which typically requires a finer resolution than is required to resolve the turbulent flow. This variation in the range of requisite spatial resolutions suggests some form of adaptive mesh refinement (AMR) to locally balance resolution with computational requirements. For the AMR methodology, we use a block-structured hierarchical form of refinement that was first developed by Berger and Oliger [29] for hyperbolic partial differential equations. A conservative version of this methodology for gas dynamics was developed by Berger and Colella [30] and extended to three dimension by Bell et al. [31]. This approach was extended to variable-density incompressible flow by Almgren et al. [32]. Pember et al. [27] generalized the approach to low Mach number combustion with simplified chemistry and tranport. The approach discussed here follows the algorithm in Day and Bell [28], which treats detailed chemistry and transport.

AMR is based on a sequence of nested grids with successively finer resolution in both time and space. In this approach, fine grids are formed by dividing coarse cells by a refinement ration, r, in each direction. Increasingly finer grids are recursively embedded in coarse grids until the solution is adequately resolved with each level contained in the next coarser level. An error estimation procedure based on user-specified criteria evaluates where additional refinement is needed and grid generation procedures dynamically create or remove rectangular fine grid patches as resolution requirements change.

The adaptive time-step algorithm advances grids at different levels using time steps appropriate to that level based on CFL considerations. The time-step procedure can most easily be thought of as a recursive algorithm, in which to advance level ℓ, $0 \le \ell \le \ell_{max}$ the following steps are taken:

- Advance level ℓ in time as if it is the only level. Supply boundary conditions for U, ρ, Y_m, h and π from level $\ell - 1$ if level $\ell > 0$, and from the physical domain boundaries.
- If $\ell < \ell_{max}$
  - Advance level $(\ell + 1)$ r times with time step $\Delta t^{\ell+1} = \frac{1}{r}\Delta t^{\ell}$ using boundary conditions for U, ρ, Y_m, h and π from level ℓ, and from the physical domain boundaries.
  - Synchronize the data between levels ℓ and $\ell+1$, interpolate corrections to higher levels if $\ell + 1 < \ell_{max}$.

The adaptive algorithm, as outlined above, performs operations to advance each level independent of other levels in the hierarchy (except for boundary conditions) and then computes a correction to synchronize the levels. (Implicit discretization must be solved over the entire level simultaneously however.) Loosely speaking, the objective in this synchronization step is to compute the modifications to the coarse grid that reflect the change in the coarse grid solution

from the presence of the fine grid. There are two steps in the synchronization. First, the fine grid is averaged onto the coarse grid; i.e., the conserved quantities on coarse grid cells covered by fine grid are replace by the average of the fine grid.

The second step of the synchronization corrects for errors at the coarse-fine boundary that arise from advancing the levels independently. A complete exposition of the details of the AMR synchronization is beyond the scope of this paper. Here we only sketch the basic ideas, see [28, 32] for details. When we advance the finer levels, we impose Dirichlet boundary conditions defined by the coarse grid at the (nonphysical) boundaries of the fine grids. As a result, the mismatch in solutions between levels ℓ and $\ell + 1$ when they reach the same point take the form of flux mismatches at the coarse/fine interface. The first step in correcting these mismatches is to define what is meant by the solution on the grid hierarchy. Using this definition, we can then computed the errors that arise from solving on each level "independently". We can then solve an appropriate correction equation to correct the solution.

The correction equations match the structure of the type of equation they are correcting. In particular, they reflect the original discrete form of the equation with a right hand side that is suppored at the coarse-fine boundary. For explicit discretizations of hyperbolic PDE's the correction is an explicit flux correction localized at the coarse/fine interface. For an elliptic equation (e.g., the projection) the source is localized on the coarse/fine interface but an elliptic equation is solved to distribute the correction through the domain. Thus the correction takes the form of a discrete analog of a layer potential problem.

The synchronization for a multiphysics algorithm such as the low Mach number projection algorithm is more complex. Essentially, we accumulate the flux mismatch that arises at each step of the algorithm and perform synchronization steps that mimic the basic steps of the algorithm. The resulting AMR algorithm then preserves the second-order accuracy and conservation properties of the underlying base discretization.

4. Software and solvers

To simulate realistic premixed turbulent flames, we must be able to implement the adaptive low Mach number algorithm described above so that we can effectively utilize high-performance parallel computers. Before discussing the implementation in more detail, we first discuss the impact of some of the choices we made in developing the basic algorithm on the design of the software. Our basic discretization strategy decomposes the problem into different mathematical components to treat advection, diffusion, chemical reactions, and projections. We use an explicit treatment of advection so that the implicit solves needed for diffusion and the projection represent discrete approximations to self-adjoint elliptic partial differential equations. Consequently, we can solve the requisite linear systems using multigrid iterative methods. Also, we have decomposed the dynamics so that the chemistry is advanced independently of the other processes. As a result the chemistry can be treated locally on a point-by-point basis.

Another important feature for the software design is the choice of AMR strategy. By adopting a block-structured form of AMR, the solution at each level in the hierarchy is naturally represented in terms of data defined on a collection of logically rectangular grid patches each containing a large number of points. Thus, the data is represented by a modest collection of relatively large regular data objects as compared to a point-by-point refinement strategy. This type of approach allows us to amortize the irregular aspects of an adaptive algorithm over large regular operations on the grid patches. This organization of data into large aggregate grid patches also provides a model for parallelization of the AMR methodology.

Our adaptive methodology is embodied in a hybrid C++/`FORTRAN` software system. In this framework, memory management and control flow are expressed in the C++ portions of the program and the numerically intensive portions of the computation are handled in `FORTRAN`. The software is written using a layered approach, with a foundation library, BoxLib, that is

responsible for the basic algorithm domain abstractions at the bottom, and a framework library, AMRLib, that marshals the components of the AMR algorithm, at the top. Support libraries built on BoxLib are used as necessary to implement application components such as interpolation of data between levels, the coarse/fine interface synchronization routines, and linear solvers used in the projections and diffusion solves.

The fundamental parallel abstraction is the `MultiFab`, which encapsulates the `FORTRAN`-compatible data defined on unions of `Box`s; a `MultiFab` can be used as if it were an array of `FORTRAN`-compatible grids. The grids that make up the `MultiFab` are distributed among the processors, with the implementation assigning grids to processors using the distribution given by the load balance scheme described in Crutchfield [33] and in Rendleman et al. [34]. This load balance scheme is based on a dynamic programming approach for solving the knapsack problem: the computational work in the irregularly sized grids of the AMR data structures is equalized among the available processors. After the initial allocation of grids some additional changes to the grid distribution are performed to reduce communications between processors. (For non-reacting flows, the number of cells per grid is often a good work estimate; for flows involving additional physics, such as chemical kinetics, the amount of work per cell is often highly variable, and work estimates based on approximating the local complexity of the kinetics are needed for good parallel performance.) `MultiFab` operations are performed with an *owner computes* rule with each processor operating independently on its local data. For operations that require data owned by other processors, the `MultiFab` operations are preceded by a data exchange between processors.

Each processor contains *meta-data* that is needed to fully specify the geometry and processor assignments of the `MultiFab`s. At a minimum, this requires the storage of an array of boxes specifying the index space region for each AMR level of refinement. In the parallel implementation, meta-data also includes the processor distribution of the `FORTRAN` compatible data. The meta-data can thus be used to dynamically evaluate the necessary communication patterns for sharing data amongst processors enabling us to optimize communications patterns within the algorithm.

5. Lean premixed hydrogen flames

The methodology describe above has been used for a number of studies of premixed methane flames, see [35–40]. Here we focus on recent work on applying the methodology to ultra-lean hydrogen combustion at conditions similar to those illustrated in Figure 2. For these preliminary simulations, rather than model the full low-swirl burner, we consider an idealized configuration in which inflowing turbulence is allowed to interact with an initially flat laminar flame. For the simulations discussed here, we use the feedback control algorithm described in [37] to stabilize the flame and consider turbulent conditions similar to what would be encountered in the central region of low-swirl burner. We also note for hydrogen we needed to increase the equivalence ratio from $\phi = 0.27$ used in the experiments to $\phi = 0.37$ in order to compensate

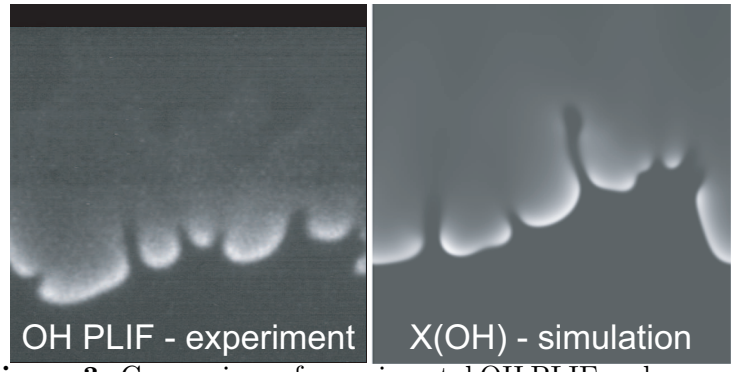


Figure 3. Comparison of experimental OH PLIF and compute OH mole fraction

for the detailed hydrogen mechanism's poor prediction of the laminar flame speed at ultra lean conditions. The fundamental issue here stems from the instability of hydrogen flames at lean conditions. In particular, at $\phi = 0.27$ it is not possible to stabilize an unstrained, flat laminar flame. Consequently, the laminar flame speed cannot be measured directly, but must be extrapolated from strained flame data. With these caveats, available mechanims appear to severely underestimate the flame speed in the ultra-lean regime. (Note that in the discussion below when we refer to a flat laminar flame, it is a theoretical construct; the real flame at these conditions is unstable.)

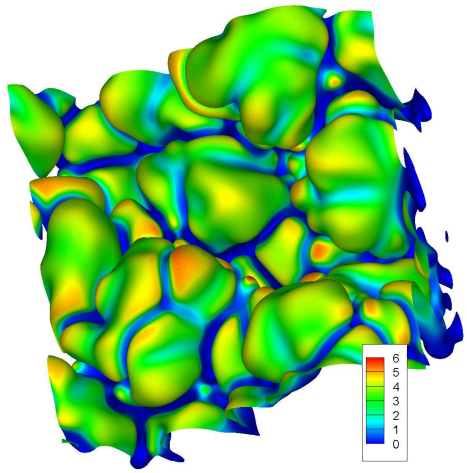


Figure 4. Isosurface of isotherm correponding to peak fuel consumption in laminar flame (1144K), colored by fuel consumption normalized by peak laminar flame value.

The simulations were performed in a domain $3 \times 3 \times 9$ cm in size with an effective resolution of 58 μm and are part of a larger study for a variety of different fuels and turbulence conditions. In Figure 3 we show representative slices from experiment and simulation for a weak turbulence case corresponding to $u' \approx 21$ cm/s showing regions of local extinction. The comparison suggests that the use of the control strategy to mimic conditions in the central region of the low-swirl burner provides a reasonable representation of the flow.

We now turn to a more detailed examination of a stronger turbulence case with $u' \approx 43$ cm/s. In Figure 4 we show an image of the instantaneous flame surface that illustrates the topology of the extinction regions, indicated by dark blue, in three dimensions. We also note that on much of the surface, the fuel consumption is considerably higher than the corresponding rate for the flat laminar flame. We also note that the extinction regions are quite robust. Once formed, they do not subsequently disappear. Instead, existing extinction regions merge together while new regions are formed as turbulence interacts with the flame.

To provide a more detailed characterization of the structure of the flame, we want to examine the flame geometry. However, to explore the geometry, we must first define the flame surface. This poses something of a problem as illustrated in Figure 5, which shows two isotherms superimposed on a slice through the flame that is colored by fuel consumption. A natural choice to define the flame surface is the $T = 1144$K isotherm, which corresponds to the temperature of peak fuel consumption in the flat laminar flame. However, because of the local extinctions this isotherm meanders through the products to bridge across gaps in the fuel consumption profile. The $T = 650$K contour reflects what would be measured experimentally using Mie scattering of oil droplets that evaporate at 650K to image the flame location. That contour does not take such severe excursions, but also does not track the fuel consumption as accurately as the $T = 1144$K contour. To provide a better definition of the flame, we define

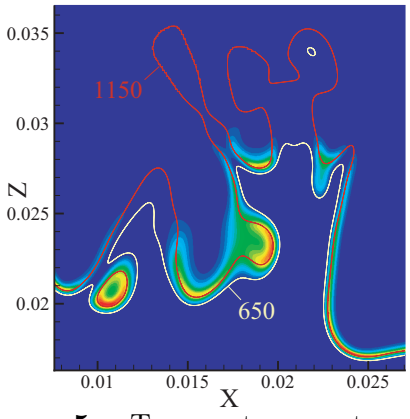


Figure 5. Temperature contours, $T = 1144$K and $T = 650$K, superimposed on slice of the fuel consumption rate through the flame.

the flame conditionally in terms of the $T = 1144$K isotherm with the added condition that the heat release on the surface be at least 10% of the laminar flame value. For a methane flame, the geometry is relatively insensitive to the precise definition of the flame; for hydrogen the differences can be substantial. For example, the instanteous flame area corresponding to the data in Figure 5 varies from 29.7 cm^2 at $T = 650$K to 45.2 cm^2 at $T = 1144$K. The conditioned data gives an area of 34.4 cm^2, which is a more realistic estimate but still somewhat arbitrary because of the choice of conditioning parameter. We also note that with the conditional definition of the flame, there is not a continuous flame surface; instead, the flame is represented as a collection of disconnected patches, or cellular structures.

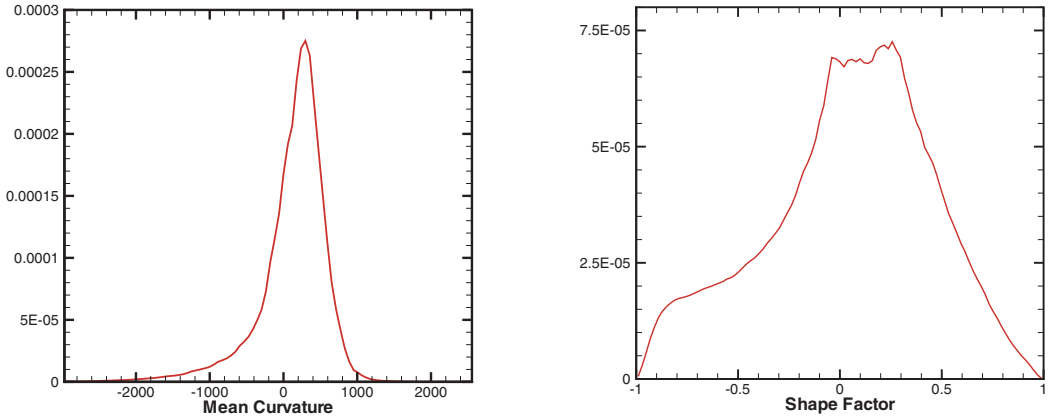


Figure 6. PDF of mean curvature and shape factor on the condition $T = 1144K$ surface.

To quantify the geometry of the flame surface, we compute mean and Gaussian curvature on the conditioned isosurface. Following Pope et al. [41], we define the shape factor, S.

$$S = \begin{cases} R_1/R_2 & \text{if } |R_1| < |R_2| \\ R_2/R_1 & \text{otherwise} \end{cases}$$

The shape factor qualitatively describes the surface: values near -1 indicate saddle points, while those near $+1$ indicate spherical regions. In Figure 6 we present PDF's of mean curvature and shape factor. The peak value of the mean curvature PDF positive and the distribution is highly asymmetric, showing that most of the flame surface is convex with respect to the unburnt gases. The shape factor, which is sharply peaked near zero for a methane flame, is also highly asymmetric, showing that much of the burning is occuring in regions where the flame is locally ellipsoidal.

Flame morphology for ultra-lean H_2 is distinctly different than a methane flame. The next question to ask is whether there are similar substantial differences in the chemical behavior of the flame. To address this question we show in Figure 7 a joint pdf of temperaure and H_2 computed from the simulation data. For comparison, we also show the analogous distribution for a methane flame. In the methane flame the temperature and CH_4 are highly correlated showing that the internal structure of the turbulent flame closely matches that of the laminar flame. In essence, the turbulence does not disturb the flame structure and the behavior is well-described by flamelet theory. The structure of the hydrogen flame is completely different. The H_2 is broadly distributed showing that the turbulence has substantially disrupted the flame. In fact, the peak of the distribution forms a ridge that is significantly shifted from the laminar flame solution. This shows that the principle mode of burning in the flame is occurring at substantially different conditions than the corresponding laminar flame; in particular, most of the burning is

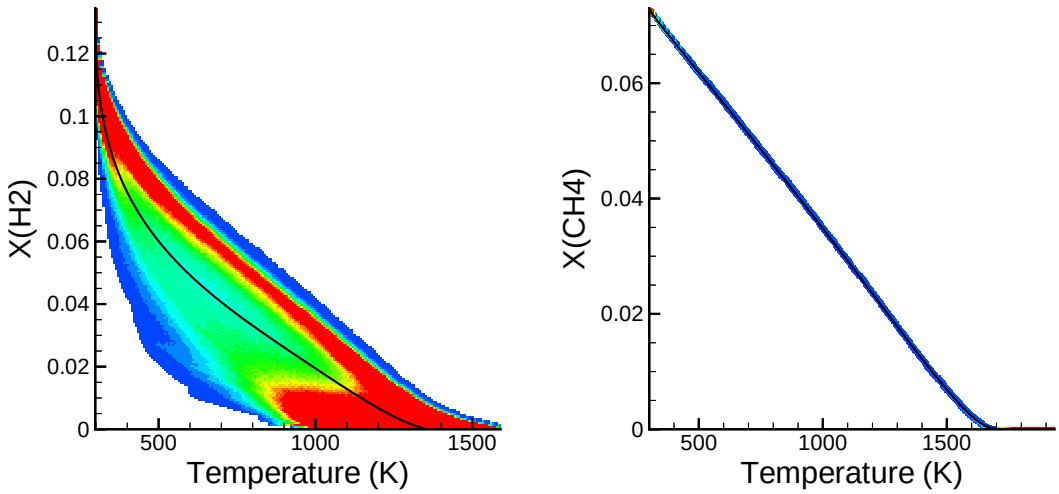


Figure 7. Joint PDF of temperature and H_2. For comparison, we present a comparable joint PDF for a methane flame. In both cases, the black line represents fuel versus temperature for a corresponding flat laminar flame.

occurring at effectively richer conditions than the inflow equivalence ratio, consistent with the enhanced fuel consumption shown in Figure 4. Preliminary analysis shows that as fluid flows through the gaps in the flame, the fuel diffuses out of those regions into the adjacent strongly burning regions enriching the combustion.

The results discussed above represent only the initial stages in understanding the dynamics of ultra-lean hydrogen flames. However, they do serve to underscore a number of feature of these flames. In particular, the flame burns in cellular structures that a surrounded by a network of extinction regions. Consequently, the traditional theoretical underpinning for the analysis of premixed flames, which assumes a continous flame sheet separating burned and unburned regions, does not apply to these flames. Analysis of the chemical behavior of the system shows that the principle mode of burning has been significantly modified and is distinctly different than that of the corresponding laminar flame. From an experimental perspective, this poses significant difficulties. Measurements can provide an estimate of the $T = 650$K isotherm and can demonstrate the existence of local extinction regions on the flame surface. However, most of the diagnostics we have considered here are simply not possible to measure and standard approaches for analysis of flame data cannot be applied. Substantial progress in our understanding of lean hydrogen combustion will only be possible by a combination of experiment and high-fidelity simulations.

6. Conclusions

In this paper, we have discussed the development of a parallel, adaptive low Mach number simulation capability for modeling premixed turbulent combustion. The methodology is an outgrowth of considering the entire process from developing a mathematical model that exploits specific features of the problem to the development of discretizations that reflect the mathematical structure of the model to software implementation issues that enable the methodology to effectively utilize high-performance parallel computers. The careful synthesis of these different elements has allowed us to construct new computational tools that are several orders of magnitude faster than more traditional approaches. This enhanced capability has made it possible to simulate realistic turbulent flames without using explicit models for turbulence or

turbulence / chemistry interaction, and we are currently working with experimentalist to use these tools to address fundamental questions in turbulent premixed combustion.

Acknowledgments
Computations were performed at the National Energy Research Scientific Computing Center, at the National Center for Computational Sciences and at the NASA Columbia. This work was supported by the DOE Office of Science through the SciDAC program by the Office of Advanced Scientific Computing Research, Mathematical, Information, and Computational Sciences Division.

References

[1] R. K. Cheng and I. G. Shepherd. The influence of burner geometry on premixed turbulent flame propagation. *Combust. Flame*, 85:7–26, 1991.

[2] S. S. Sattler, D. A. Knaus, and F. C. Gouldin. Determination of three-dimensional flamelet orientation distributions in turbulent V-flames from two-dimensional image data. *Proc. Combust. Inst.*, 29:1785–1795, 2002.

[3] I. G. Shepherd, R. K. Cheng, T. Plessing, C. Kortschik, and N. Peters. Premixed flame front structure in intense turbulence. *Proc. Combust. Inst.*, 29:1833–1840, 2002.

[4] D. Most, F. Dinkelacker, and A. Leipertz. Lifted reaction zones in premixed turbulent bluff-body stabilized flames. *Proc. Combust. Inst.*, 29:1801–1806, 2002.

[5] Y.-C. Chen, P. A. M. Kalt, R. W. Bilger, and N. Swaminathan. Effects of mean flow divergence on turbulent scalar flux and local flame structure in premixed turbulent combustion. *Proc. Combust. Inst.*, pages 1863–1871, 2002.

[6] N. Peters. *Turbulent Combustion*. Cambridge University Press, Cambridge, 2000.

[7] K. N. C. Bray. The challenge of turbulent combustion. *Proc. Combust. Inst.*, 26:1–26, 1996.

[8] Y. C. Chen and R. W. Bilger. Experimental investigation of three-dimensional flame-front structure in premixed turbulent combusiton II. Lean hydrogen/air Bunsen flames. *Combust. Flame*, 138:155–174, 2004.

[9] H. G. Im and J. H. Chen. Preferential diffusion effects on the burning rate of interacting turbulent premixed hydrogen-air flames. *Combust. Flame*, 131:246–258, 2002.

[10] R. J. Kee, J. Warnatz, and J. A. Miller. A FORTRAN computer code package for the evaluation of gas-phase viscosities, conductivities, and diffusion coefficients. Technical Report SAND83-8209, Sandia National Laboratories, 1983.

[11] J. Warnatz. Influence of transport models and boundary conditions on flame structure. In N. Peters and J. Warnatz, editors, *Numerical Methods in Laminar Flame Propagation*, pages 87–111. Vieweg-Verlag, Braunschweig, 1982.

[12] R. G. Rehm and H. R. Baum. The equations of motion for thermally driven buoyant flows. *N. B. S. J. Res.*, 83:297–308, 1978.

[13] A. Majda and J. A. Sethian. The derivation and numerical solution of the equations for zero Mach number combustion. *Combust. Sci. Technol.*, 42:185–205, 1985.

[14] K. E. Brenan, S. L. Campbell, and L. R. Petzold. *Numerical Solution of Initial-Value problems in Differential-Algrebraic Equations*. SIAM, Philadelphia, PA, 1996.

[15] U. Ascher and L. R. Petzold. Projected implicit Runge Kutta methods for differential algebraic systems. *SIAM J. Num. Anal.*, 28:1097–1120, 1991.

[16] A. J. Chorin. Numerical solution of the Navier-Stokes equations. *Math. Comp.*, 22:745–762, 1968.

[17] J. B. Bell, P. Colella, and H. M. Glaz. A second-order projection method for the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 85(2):257–283, December 1989.

[18] A. S. Almgren, J. B. Bell, and W. G. Szymczak. A numerical method for the incompressible Navier-Stokes equations based on an approximate projection. *SIAM J. Sci. Comput.*, 17(2):358–369, March 1996.

[19] P.A. McMurtry, W.-H. Jou, J.J. Riley, and R.W. Metcalfe. Direct numerical simulations or a reacting mixing layer with chemical heat release. *AIAA J.*, 24:962, 1986.

[20] C.J. Rutland and J.H. Ferziger. Simulations of flame-vortex interactions. *Combust. Flame*, 84:343, 1991.

[21] S. Zhang and C. J. Rutland. Premixed flame effects on turbulence and pressure-related terms. *Combust. Flame*, 102:447–461, 1995.

[22] H. N. Najm and P. S. Wyckoff. Premixed flame response to unsteady strain rate and curvature. *Combust. Flame*, 110(1–2):92–112, 1997.

[23] H. N. Najm, O. M. Knio, P. H. Paul, and P. S. Wyckoff. A study of flame observables in premixed methane-air flames. *Combust. Sci. Technol.*, 140:369–403, 1998.

[24] H. N. Najm, P. S. Wyckoff, and O. M. Knio. A semi-implicit numerical scheme for reacting flow. I. Stiff chemistry. *J. Comput. Phys.*, 143:381–402, 1998.

[25] J. Qian, G. Tryggvason, and C. K. Law. Front tracking method for the motion of premixed flames. *J. Comput. Phys.*, 144:52–69, 1988.

[26] J. B. Bell and D. L. Marcus. A second-order projection method for variable density flows. *J. Comput. Phys.*, 101(2):334–348, August 1992.

[27] R. B. Pember, L. H. Howell, J. B. Bell, P. Colella, W. Y. Crutchfield, W. A. Fiveland, and J. P. Jessee. An adaptive projection method for unsteady, low-Mach number combustion. *Combust. Sci. Technol.*, 140:123–168, 1998.

[28] M. S. Day and J. B. Bell. Numerical simulation of laminar reacting flows with complex chemistry. *Combust. Theory Modelling*, 4:535–556, 2000.

[29] M. J. Berger and J. Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *J. Comput. Phys.*, 53:484–512, March 1984.

[30] M. J. Berger and P. Colella. Local adaptive mesh refinement for shock hydrodynamics. *J. Comput. Phys.*, 82(1):64–84, May 1989.

[31] J. Bell, M. Berger, J. Saltzman, and M. Welcome. A three-dimensional adaptive mesh refinement for hyperbolic conservation laws. *SIAM Journal on Scientific and Statistical Computing*, 15(1):127–138, 1994.

[32] A. S. Almgren, J. B. Bell, P. Colella, L. H. Howell, and M. L. Welcome. A conservative adaptive projection method for the variable density incompressible Navier-Stokes equations. *J. Comput. Phys.*, 142:1–46, May 1998.

[33] W. Y. Crutchfield. Load balancing irregular algorithms. Technical Report UCRL-JC-107679, Lawrence Livermore National Laboratory, July 1991.

[34] Charles A. Rendleman, Vincent E. Beckner, Mike Lijewski, William Y. Crutchfield, and John B. Bell. Parallelization of structured, hierarchical adaptive mesh refinement algorithms. *Computing and Visualization in Science*, 3(3):147–157, 2000.

[35] J. B. Bell, N. J. Brown, M. S. Day, M. Frenklach, J. F. Grcar, and S. R. Tonse. The dependence of chemistry on the inlet equivalence ratio in vortex-flame interactions. *Proc. Combust. Inst.*, 28:1933–1939, 2000.

[36] J. B. Bell, M. S. Day, and J. F. Grcar. Numerical simulation of premixed turbulent methane combustion. *Proc. Combust. Inst.*, 29:1987–1993, 2002.

[37] J. B. Bell, M. S. Day, J. F. Grcar, and M. J. Lijewski. Active control for statistically stationary turbulent premixed flame simulations. *Comm. App. Math. Comput. Sci.*, 1:29–52, 2006.

[38] J. B. Bell, M. S. Day, I. G. Shepherd, M. Johnson, R. K. Cheng, J. F. Grcar, V. E. Beckner, and M. J. Lijewski. Numerical simulation of a laboratory-scale turbulent V-flame. *Proc. Natl. Acad. Sci. USA*, 102(29):10006–10011, 2005.

[39] J. B. Bell, M. S. Day, J. F. Grcar, M. J. Lijewski, J. F. Driscoll, and S. A. Filatyev. Numerical simulation of a laboratory-scale turbulent slot flame. *Proc. Combust. Inst.*, 31, 2006. to appear.

[40] J. B. Bell, M. S. Day, R. K. Cheng, and I. G. Shepherd. Numerical simulation of lewis number effects on lean premixed turbulent flames. *Proc. Combust. Inst.*, 31, 2006. to appear.

[41] S. B. Pope, P. K. Yeung, and S. S. Girimaji. The curvature of material surfaces in isotropic turbulence. *Phys. Fluids A*, 1:2010–2018, 1989.